Sarah Sandstrom

December 9, 2021

CS-320

Project 2

When I was working on the project I knew I needed to test each section individually before I could know if they would work together. For the requirements I knew that with the appointment testing section I needed to test the entire area of information needed to make an appointment. I needed to make sure that the appointments had everything they needed before the test even began, such as the id number, the description of what the appointment is for, and for the date all can be seen in lines 16-23 of AppointmentServiceTest.java. Which then runs through each line to make sure everything is filled out and has the right information which is included in the test portion from lines 25-55. Then we have the test portion to delete an appointment in lines 57-81. This makes it so we can delete the entire appointment. In AppointmentTest.java I focused on each individual area needed so if we needed to update or get the appointment id, date, and description.

When looking at ContactServiceTest.java I had the before section just like in AppointmentServiceTest.java where all the information I needed for the contact information such as, the contact Id, first name, last name, phone number, and address, as well as, if their contact information was too long for parameters how the system would then save their information. Then I tested the new contact in lines 29-118. Then I had the delete contact from lines 120-128. Then I went on to change any of the information like the contacts first name, last name, phone number, or their address. With the ContactTest.java I was making sure that all of the information needed was filled out.

For TaskServiceTest.java I did the same thing as the two before where at the beginning I made sure to put the parameters of what information would be needed so for tasks we need the id, name, and description. Then I added new task, task name, and task description. Then I tested if the name or description was too long. I also tested to make sure that nothing was left empty that needed to be filled out. Then I did the delete task as well as updating the task name and description. Then I added for the test to be repeated shown in line 128-141. With TaskTest.java I focused on us getting the task id, name, and description, as well as to set the name and description. I added in the task id, name, or description was too long. And the null test to make sure nothing was missing.

I made sure that the code was to work together at the end. I wanted to make sure that all the input would then output the same way unless the input did not fill the parameters I set. I made sure that with every input there would be a test to make sure it fit within the parameters. What I did was break it down into sections, appointment, contact, and task. Then I went on and wrote the code for appointment and appointment service then the test section for appointment service test and appointment test. And then I did the same with contact and task. I wanted to make sure that everything that had parameters that I set in the main section was covered in the test section. To make sure my code was working once I finished a section of code, I would run the code to see if any errors occurred, but also would it show if I input something not within the parameters would it notify me. I made sure to do so with every section.

For the software testing techniques, I used in this project was white box testing. I did not used this because even though I knew what kind of structure I wanted and then implemented and I wanted to start testing in the early stages instead of later. I did not have a control flow or data flow testing. I did however use black box because I made the testing portion focus mainly on incorrect or missing functions, and performance errors. I also used agile testing because I constantly was testing my code to make sure everything was running smoothly.

The mindset I had when working on this project was to take it one step at a time. When I first saw the different sections of code, I needed to write I was overwhelmed at first. Then I decided to break it down into sections so that why when writing code, I was only focusing on one section. Once I was done with one section like appointments I then tested to see if I was missing anything or if anything would go wrong. I continued that process for contact and tasks sections as well. I think breaking it down made me less stressed but also helped me perform my code better because I was focused on one task at a time. You can see that throughout my code I had made the main section of appointments first and then went into the test section and then tested the appointments to make sure all the parameters were working. I think this was a good way to do it because I did run into a few problems at the beginning and then once those were fixed and I went onto the sections for contact and tasks I did not make the same mistakes as before. If I had just done it all in one then I would have probably made the same mistake over again.